WEST

Generate Collection | Print

Search Results - Record(s) 1 through 1 of 1 returned.

□ 1. Document ID: US 6167400 A

L6: Entry 1 of 1 File: USPT Dec 26, 2000

US-PAT-NO: 6167400
DOCUMENT-IDENTIFIER: US 6167400 A

TITLE: Method of performing a sliding window search

DATE-ISSUED: December 26, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Brandin; Christopher Lockton	Colorado Springs	CO		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
NEO-Core					02

APPL-NO: 9/ 334043
DATE FILED: June 15, 1999

PARENT-CASE:
This application claims benefit of provisional application 60/094,968, filed Jul. 31, 1998.

INT-CL: [7] G06 F 7/02

US-CL-ISSUED: 707/6
US-CL-CURRENT: 707/6

FIELD-OF-SEARCH: 707/3, 707/5, 707/6, 708/209

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
5051947	September 1991	Messenger et al.	707/3
5497488	March 1996	Akizawa et al.	707/6

ART-UNIT: 277
PRIMARY-EXAMINER: Kulik; P V.
ATTY-AGENT-FIRM: Halling; Dale B.

ABSTRACT:

A method of performing a sliding window search includes the steps of: (a) creating an associative database of a plurality of data strings; (b) receiving a first window of a data block; (c) iconizing the first window of the data block to form a first icon; (d)

determining if the first icon has a match in the associative database; (e) determining a first byte icon of a first byte of data in the first window; (f) executing an icon shift function to form a shifted first byte icon; (g) exclusive ORing the shifted first byte icon with the first icon to form a seed icon; (h) determining a second icon for a second window using the seed icon and transforming a new byte of data onto the seed icon; and (i) determining if the second icon has a match in the associative database.

21 Claims, 12 Drawing figures

Full | Title CIT.1 REV.1 CLS.1 REF.1 SEQ.1 ATT.1

RAW.1

Generate Collection | Print

Terms	Documents
associative adj database.clm.	1

Display Format: FRO | Change Format

Previous Page Next Page

WEST

End of Result Set

□ Generate Collection Print

L6: Entry 1 of 1 File: USPT Dec 26, 2000

US-PAT-NO: 6167400
DOCUMENT-IDENTIFIER: US 6167400 A

TITLE: Method of performing a sliding window search

DATE-ISSUED: December 26, 2000

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Brandin; Christopher Lockton	Colorado Springs	CO		

US-CL-CURRENT: 707/6

CLAIMS:

What is claimed is:

1. A method of performing a sliding window search, comprising the steps of:

(a) creating an associative database of a plurality of data strings;

(b) receiving a first window of a data block;

(c) iconizing the first window of the data block to form a first icon;

(d) determining if the first icon has a match in the associative database;

(e) determining a first byte icon of a first byte of data in the first window;

(f) executing an icon shift function to form a shifted first byte icon;

(g) exclusive ORing the shifted first byte icon with the first icon to form a seed icon;

(h) determining a second icon for a second window using the seed icon and transforming a new byte of data onto the seed icon;

and

(i) determining if the second icon has a match in the associative database.

2. The method of claim 1, wherein step (a) further includes the steps of:

(a1) determining if a single search window size is required;

(a2) when the single search window size is required, determining an icon for each of the plurality of data strings.

3. The method of claim 2, further including the steps of:

(a3) when more than the single search window size is required, determining a minimum length search window;

(a4) determining an icon for each of a first plurality of data strings having a length equal to the minimum length to form a plurality of first icons;

(a5) storing the plurality of first icons in the associative database.

4. The method of claim 3, further including the steps of:

(a6) determining an icon for a first portion of each of a second plurality of data strings to form a plurality of second icons;

(a7) storing the plurality of second icons in the associative database.

5. The method of claim 4, further including the steps of:

(a8) determining an icon for a second portion of each of the second plurality of data strings to form a plurality of third icons;

(a9) storing the plurality of third icons in the associative database;

(a10) storing a pointer with each of the plurality of second icons, the pointer pointing to one of the plurality of third icons.

6. The method of claim 5, wherein step (d) further includes the steps of:

(d1) when the first icon is found in the associative database, determining if the pointer is stored with the first icon;

(d2) when the pointer is not stored with the first icon, determining the match has been found.

7. The method of claim 6, further including the steps of:

(d3) when the pointer is stored with the first icon, determining a next icon;

(d4) comparing the next icon to an icon at a pointer location;

(d5) when the next icon is the same as the icon at the pointer location, determining the match has been found.

8. The method of claim 3, further including the steps of:

(d1) when the first icon is found in the associative database and includes a pointer, determining a second icon;

(d2) determining if the second icon has a match in the associative database.

9. The method of claim 8, wherein the step of determining a second icon includes performing an icon append operation with a second portion to the first icon.

10. A method of performing a sliding window search, comprising the steps of:

(a) generating an associative database;

(b) selecting a first window of a data block to be examined;

(c) iconizing the first window to form a first icon;

(d) performing a lookup in the associative database to determine if there is a match;

(e) selecting a second window of the data block, wherein the second window contains a new portion and a common portion of the first window;

(f) determining a second icon using the first icon, a discarded portion and the new portion but not the common portion, the second icon being associated with the second window; and

(g) performing a lookup in the associative database using the second icon to determine if there is a match.

11. The method of claim 10, further including the step of:

(h) returning to step (d).

12. The method of claim 10, wherein step (c) further includes the step of:

(c1) performing a linear feedback shift register operation on the first window to create the first icon.

13. The method of claim 12, wherein the step of performing the linear feedback shift register operation is a cyclical redundancy code.

14. The method of claim 10, wherein step (f) further includes the steps of:

(f1) determining a discarded icon for the discarded portion;

(f2) executing an icon shift function to form a shifted discarded icon;

(f3) exclusive ORing the shifted discarded icon with the first icon to form a seed icon.

15. The method of claim 14, further including the steps of:

(f4) determining a new icon for the new portion;

(f5) exclusive ORing the new icon with the seed icon to form the second icon.

16. The method of claim 10, wherein step (d) further includes the steps of:

(d1) determining if the associative database indicates a match, a no match or a qualifier match;

(d2) when a qualifier match is indicated, determining a next window icon for a next complete non-overlapping window of the data block;

(d3) determining if there is a link pointer, pointing from the first icon to the next window icon.

17. The method of claim 10, wherein step (d) further includes the steps of:

(d1) determining if the associative database indicates a match, a no match or a qualifier match;

(d2) when a qualifier match is indicated, determining a match length;

(d3) appending an extra portion onto the first icon to form a second icon;

(d4) determining if the associative database indicates a match.

18. A method of performing a sliding window search, comprising the steps of:

(a) selecting a plurality of data strings to be found;

(b) iconizing each of the plurality of data strings to form a plurality of match icons;

(c) creating an associative database having a plurality of addresses, wherein each of the plurality of match icons corresponds to one of the plurality of addresses; and

(d) storing a match flag at each of the plurality of addresses corresponding to the plurality of match icons.

19. The method of claim 18, further including the steps of:

(e) when the plurality of data strings do not all have a same length, selecting a plurality of shortest data strings;

(f) determining a plurality of short icons and storing a match indicator in the associative database;

(g) determining a plurality of qualifier icons for a first portion of a plurality of longer data strings;

(i) storing a qualifier flag for each the plurality of qualifier icons in the associative database.

20. The method of claim 19, further including the step of:

(j) storing a match length indicator at each of the plurality of qualifier icons in the associative database.

21. The method of claim 20, further including the steps of:

(k) determining an icon for a first window of a data block, wherein the first window has a window length equal to a shortest length;

(l) performing a lookup in the associative database to determine if there is the match flag or the qualifier flag;

(k) when there is the qualifier flag, retrieving the match length indicator;

(l) determining a complete icon for a portion of the data block equal to the match length;

(m) performing a lookup in the associative database to determine if there is a match flag associated with the complete icon.